**CACIE Tool #02** – ***Kingdom2Stomp Tool  
K2S\_ROCSAN.exe***

**Version** **1.0**

**QA**

1. **Description and Purpose**

The Kingdom2Stomp Tool reads a comma-separated values (CSV) input file representing each node in the model and generates a CSV output file like the input file with the addition of what geologic formation each model node represents. This tool assigns geologic formations to the STOMP grid points using the formation top information developed typically by Kingdom but could also come from ArcGIS (separate licensed programs). The output file generated from Kingdom2Stomp will be used as another input file to the CAST program for mapping the geologic materials to the user-defined STOMP model boundaries.

An additional feature is the capability to read well locations and extract well data that is within the user-defined STOMP grid boundaries. This feature is titled ROCSAN. It searches the ROCSAN files using the extracted well information for sand/gravel/silt details. It finds the formation identity at each sample depth and writes all samples found for each formation in that STOMP node. This feature is currently not used in the Composite Analysis (CA) project and will not be quality assurance (QA) qualified in this document.

1. **Functional Requirements**

The following are the functional requirements (FR) of the Kingdom2Stomp Tool:

FR-1: Command Line Argument to specify the configuration file.

FR-2: Ability to process a user-provided configuration file that specifies geology *\*.dat* files and site specific X, Y, Z gridding values from CSV files.

FR-3: Open and read the geology *\*.dat* (created by Kingdom or ArcGIS) files listed at the top of the configuration input file.

FR-4: Read and store the X, Y, and Z gridding values from the *ModelName.csv* file (provided in the configuration file after the geology dat files), where “ModelName” is the name of the STOMP model.

FR-5: Read from the configuration file, below the “ModelName.csv” line(s), to assign a formation name to each STOMP grid block.

FR-6: Create an output file name based on the user defined name (specified in the last line of the configuration file) and assign a formation for each STOMP grid block from the input file.

1. **Software Requirements Specifications**

Programming Language and required modules/libraries for the Kingdom2Stomp tool are as follows:

VisualBasic

1. **Software Design Description**

The following are required arguments for the Kingdom2Stomp execution, necessary input files, and output files generated.

Arguments:

Position 1: Configuration file name/location

Input Files:

Configuration file – which will have the name and path to various inputs and outputs in the below format:

* + Line 1 indicating the number of geology *\*.dat* files that follow.
    - The number of lines is dependent on location in the Central Plateau, either 11 lines for 200W or 14 lines for 200E.
  + Line 2-X specify the path and name of each geology file.
    - The first line is Surface Deposits and the last will be Basalt. All other files are in between.
  + First line after the geology files will state “1”, indicating the number of CSV files in the following line(s).
  + The following line points to the *csv* file created by CAST.
  + The following line says “yes”, “no”.
    - The first item in this line identifies if Kingdom2Stomp executes or not.
    - The second item in this line identifies if ROCSAN executes or not.
  + The next line is the Kingdom2Stomp output file title, example: *test1-eval.csv*.
  + The last line should read zero, “0”.

Output File:

This file will contain the same first three columns the input csv (from CAST) file contains, with the addition of a fourth column, geologic formation.

Tool Runner:

The following is the shell script configuration that will be passed as an argument to the Tool Runner for qualified runs:

{directory path to repository}\Kingdom2Stomp\Kingdom\EXE\K2S\_ROCSAN.exe {configuration file path/name}

Code Review:

There is no documentation linking the current version of the code to the executable that is currently in use. As we cannot verify the code belongs to the current executable, we will be deferring the code review until the next revision.

1. **Requirements Traceability Matrix**

The Requirements Traceability Matrix for the Kingdom2Stomp tool is presented in Table 1.

| **Table 1 Requirements Traceability Matrix** | | |
| --- | --- | --- |
| **Functional Requirement ID** | **Acceptance Test ID** | **Test Case** |
| QA Level | CACIE-K2S\_ROCSAN.exe -IT-1 | Installation Test |
| FR-1 | CACIE-K2S\_ROCSAN.exe-TC-1  CACIE-K2S\_ROCSAN.exe-TC-2 | Verify the *runner\_runK2S\_ROCSAN.bat* file is pointing to the correct executable directory and the correct configuration input *\*.txt* file. |
| FR-2 | CACIE-K2S\_ROCSAN.exe-TC-1  CACIE-K2S\_ROCSAN.exe-TC-2 | Confirm in the configuration file: |
| Line 1 states “14”, indicating the number of geology *\*.dat* files that follow. |
| Lines 2 through 15 are the geology *\*.dat* files for each formation. |
| Line 16 indicates the number of model CSV ModelName files and the following number of lines point to each “*ModelName.csv*” file. |
| The following line states “yes”, “no”. |
| The next line or lines will be the file output title, “*ModelName-eval.csv*” for each ModelName input file. |
| The last line should read zero, “0”. |
| FR-3 | CACIE-K2S\_ROCSAN.exe-TC-1  CACIE-K2S\_ROCSAN.exe -TC-2 | Confirm all the geology *\*.dat* formation files are present in the directory identified in configuration file. |
| FR-4 | CACIE-K2S\_ROCSAN.exe-TC-1  CACIE-K2S\_ROCSAN.exe-TC-2 | Confirm the *ModelName.csv* file has three columns of data, representing X, Y, and Z, respectively. |
| FR-5  FR-6 | CACIE-K2S\_ROCSAN.exe-TC-1  CACIE-K2S\_ROCSAN.exe-TC-2 | Confirm the *ModelName-eval.csv* output file has four columns, representing X, Y, Z, and the associated geology formation, respectively. |

1. **Test Plan and Cases**

The Installation Test Plan for the Kingdom2Stomp tool is presented in Table 2 and the Acceptance Test Plan is presented in Table 3.

| **Table 2**  **Kingdom2Stomp Installation Test Plan** | | | |
| --- | --- | --- | --- |
| **Kingdom2Stomp Installation Testing**  **CACIE-** **K2S\_ROCSAN.exe–IT-1** | | **Date:** | |
| **Tool Runner File Location for this test:** | | **Test Performed By:** | |
| **Testing Directory:**  Runs in Windows | | | |
| **Test Step** | **Test Instruction** | **Expected Result** | **Test Result  (Pass/Fail)** |
| Note the Tools Code Repository: | | | |
| Navigate to the testing directory. | | | |
| 1 | Invoke Tool Runner and test the tool as follows: *./runner\_run\_IT-1\_* *K2S\_ROCSAN.exe.bat* | | |
| 2 | Verify Tool Runner is invoked and executed. |  |  |
| 3 | Verify tool is invoked and executed. |  |  |

| **Table 3**  **Kingdom2Stomp Acceptance Test Plan** | | | |
| --- | --- | --- | --- |
| **Kingdom2Stomp Acceptance Testing**  **CACIE-K2S\_ROCSAN.exe–AT-##** | | **Date:** | |
| **Tool Runner File Location for this test:** | | **Test Performed By:** | |
| **Testing Directory:**  Runs in Windows | | | |
| **Test Step** | **Test Instruction** | **Expected Result** | **Test Result  (Pass/Fail)** |
| **CACIE-** **K2S\_ROCSAN.exe-TC-1** | | | |
| Navigate to the [Testing Directory]\acceptance\_test\_1 | | | |
| Confirm in the configuration file, *test1\_configuration.txt*, Steps 1-6: | | | |
| 1 | Line 1 states “11” indicating the number of geology *\*.dat* files that follow. | 11 |  |
| 2 | Lines 2 through 12 are the geology *\*.dat* files for each formation. | First line is Surface Deposits and the last is Basalt. All other files are in between. |  |
| 3 | Line 13 states “1” and the following line points to the *test1.csv* file. | 1 “test1.csv” |  |
| 4 | The following line says “yes”, “no”. | “yes”,”no” |  |
| 5 | The next line is the output file title, *test1-eval.csv*. | “test1-eval.csv” |  |
| 6 | The last line should read zero, “0”. | 0 |  |
| 7 | Confirm all the geology *\*.dat* formation files are present in the directory identified in *test1\_configuration.txt* file. | There will be 11 files present, each corresponding to lines 2 through 12 in the configuration file. |  |
| 8 | Confirm the *test1.csv* file has three columns of data, representing X, Y, and Z, respectively. | There will be three columns of numbers in order – X, Y then Z. No header in this file. |  |
| 9 | Open a command prompt in the test directory and type in the below command to execute script: *runner\_runK2S\_ROCSAN.bat* | 2 files should be created:   * *Test1-eval.csv* * Tool Runner log |  |
| 10 | Confirm the *test1-eval.csv* output file has four columns, representing X, Y, Z, and the associated geology formation, respectively. | The same first three columns in the input file are present, with the addition of a fourth column – geologic formation. |  |
| 11 | Using the X, Y, and Z values in the output file, *test1-eval.csv*, determine if the formation written matches with the *\*.dat* formation files. Use either Excel or Surfer to accomplish this. | The geologic formations written for each X, Y, and Z row in the output file will match the *\*.dat* formation files. |  |

1. **Acceptance Test Report**

To complete the Acceptance Testing use Appendix A. The Acceptance Test case are described as follows:

* Acceptance Test 1 is in Table A-1. This acceptance test is the LW Crib Area Model in the 200E of the Central Plateau. Gridding in the X and Y direction are of variable lengths/widths. The X gridding is from 567200 m to 568800 m, the Y gridding is from 135600 m to 136400 m, and the Z gridding is from 126.9 m to 224.4 m.

Details of this test, when it was conducted, by whom, and if it Passed or Failed is present in Appendix A.

1. **User Guide**

The following is a guide for using the Kingdom2Stomp tool. For explicit information on the arguments and input file reference Section 4: Software Design Description. Before you execute this application, you will need geology files generated using Kingdom and a csv file generated from CAST containing the gird information (x,y,z coordinates) specific to your model. Once your Configuration file has been created (see Section 4: Software Design Description for format) you will be ready to start.

Execute Kingdom2Stomp from a windows command prompt. using the below command, replacing {configuration file} with the path and filename of your configuration file.

K2S\_ROCSAN.exe {configuration file}

Note, this script will take time to generate the file. Inside the command window the prompt will not allow you to type or execute anything until the script is complete.

**Appendix A**

**Acceptance Testing Logs**

**Testing Process Description:**

Test CACIE- K2S\_ROCSAN.exe-AT-1:

1. Performed CACIE- K2S\_ROCSAN.exe -AT-1 Test Steps 1 through 9 (as described in Table A-1 below) – All passed.
2. Used Golden Software Surfer 9 and Excel to assign material type to each model node (files in v4‑2\_kingdom2stomp\_acceptance\_test\_1.zip):  
   a) Gridded all 200 West Kingdom dat files using dx and dy values from the Kingdom files to define the grid node spacing (\*.grd files).  
   b) Copied test1.csv to test1\_Surfer\_redisuals.csv.  
   c) Used the Surfer grid residuals option to calculate a residual (node elevation - Kingdom surface elevation) for each model node (test1\_Surfer\_redisuals.csv).  
   d) Evaluated residuals for each surface for each node to determine the material type for each node (test1\_Surfer\_redisuals.xlsx; see “Notes” worksheet for more details about this process).  
   e) Exported a file containing X, Y, Z and material type (test1\_Surfer\_redisuals\_units.csv) from test1\_Surfer\_redisuals.xlsx.
3. Compared test1-eval.csv (output from Kingdom2Stomp) to test1\_Surfer\_redisuals\_units.csv in check\_test1-eval.xlsx (see “Notes” worksheet for more details about this process) – All material types match; CACIE-Kingdom2Stomp-AT-1 Test Step 10 passed.

**Tool Runner Log:**

**INFO--01/29/2020 02:09:01 PM--Starting CA-CIE Tool Runner. Logging to "../runner\_K2S\_ROCSAN\_test\_logfile.txt"**

**INFO--01/29/2020 02:09:01 PM--Invoking: Command:"../../CA-CIE-Tools-Testing/tools/Kingdom2Stomp/Kingdom/EXE/K2S\_ROCSAN.exe" Arguments:".\test1\_configuration.txt"**

**INFO--01/29/2020 02:09:05 PM--Code Version: 1f1673849bd63b1795ee57fe48b0a237a91663d6 v1.1: S:/PSC/!HANFORD/ICF/CA-CIE-Tools/CA-CIE-Tools/pylib/runner/runner.py**

**INFO--01/29/2020 02:09:06 PM--Code Version: ecd17a8b08df3aea37edf1d043b18dfd1fbbf53c Local repo SHA-1 has does not correspond to a remote repo release version: ../../CA-CIE-Tools-Testing/tools/Kingdom2Stomp/Kingdom/EXE/K2S\_ROCSAN.exe**

**INFO--01/29/2020 02:09:09 PM--QA Status: QUALIFIED : S:/PSC/!HANFORD/ICF/CA-CIE-Tools/CA-CIE-Tools/pylib/runner/runner.py**

**INFO--01/29/2020 02:09:10 PM--QA Status: TEST : ../../CA-CIE-Tools-Testing/tools/Kingdom2Stomp/Kingdom/EXE/K2S\_ROCSAN.exe**

**INFO--01/29/2020 02:09:10 PM--Username:dfryar Computer:PSC-Sodium Platform:Windows 10 10.0.18362**

| **Table A-1**  **Kingdom2Stomp Acceptance Test Case 1** | | | |
| --- | --- | --- | --- |
| **Kingdom2Stomp Acceptance Testing**  **CACIE-** **K2S\_ROCSAN.exe-AT-1** | | **Date: 01-29-2020** | |
| **Tool Runner File Location for this test:** [\\olive\cfarrow\CAVE\CA-CIE-Tools-TestEnv\v4-2\_kingdom2stomp\acceptance\_test\_1](file:///\\olive\cfarrow\CAVE\CA-CIE-Tools-TestEnv\v4-2_kingdom2stomp\acceptance_test_1) | | **Test Performed By: Dennis Fryar** | |
| **Testing Directory:**[\\olive\backups\CAVE\CA-CIE-Tools-TestEnv\v4-2\_kingdom2stomp\acceptance\_test\_1](file:///\\olive\backups\CAVE\CA-CIE-Tools-TestEnv\v4-2_kingdom2stomp\acceptance_test_1) Runs in Windows | | | |
| **Test Step** | **Test Instruction** | **Expected Result** | **Test Result  (Pass/Fail)** |
| Navigate to the Testing Directory | | | |
| Confirm in the configuration file, *test1\_configuration.txt*, Steps 1-6: | | | |
| 1 | Line 1 states “11” indicating the number of geology *\*.dat* files that follow. | 11 | **Pass** |
| 2 | Lines 2 through 12 are the geology *\*.dat* files for each formation. | First line is SurfaceDeposits and the last is Basalt. All other files are in between. | **Pass** |
| 3 | Line 13 states “1” and the following line points to the *test1.csv* file. | 1 “test1.csv” | **Pass** |
| 4 | The following line says “yes”, “no”. | “yes”,”no” | **Pass** |
| 5 | The next line is the output file title, *test1-eval.csv*. | “test1-eval.csv” | **Pass** |
| 6 | The last line should read zero, “0”. | 0 | **Pass** |
| 7 | Confirm all the geology *\*.dat* formation files are present in the directory identified in *test1\_configuration.txt* file. | There will be 11 files present, each corresponding to lines 2 through 12 in the configuration file. | **Pass** |
| 8 | Confirm the *test1.csv* file has three columns of data, representing X, Y, and Z, respectively. | There will be three columns of numbers in order – X, Y then Z. No header in this file. | **Pass** |
| 9 | Open command prompt in the test directory and type in: *runner\_runK2S\_ROCSAN.bat* | 2 files should be created:   * *test1-eval.csv* * Tool Runner log | **Pass** |
| 10 | Confirm the *test1-eval.csv* output file has four columns, representing X, Y, Z, and the associated geology formation, respectively. | The same first three columns in the input file are present, with the addition of a fourth column – geologic formation. | **Pass** |
| **Screenshot of the Tool Runner executed for this test.**  Text file present in the testing directory. | | | |
| 11 | Using the X, Y, and Z values in the output file, *test1-eval.csv*, determine if the formation written matches with the *\*.dat* formation files. Use either Excel or Surfer to accomplish this. | The geologic formations written for each X, Y, and Z row in the output file will match the *\*.dat* formation files. | **Pass** |